Polytomous DIF

Danny

3/2/2021

Contents

```
library(tidyverse)
library(TAM)
```

1 Expressing the PCM and RSM with formulas and facets (When number of response options same across items)

We'll use the polytomous data that we used in lab 6. It has covariates that we're interested in. We'll use the "Gender" covariate for DIF.

We have a few more options regarding DIF with polytomous data. I'll use the delta-tau parameterization to make talking about the Partial Credit Model (PCM) and Rating Scale Model (RSM) easier.

As a reminder the RSM can be expressed:

$$P(X_{si} = x) = \frac{exp[\sum_{k=0}^{x}(\theta_s - \delta_i + \tau_k)]}{\sum_{h=0}^{m_i} exp[\sum_{k=0}^{h}(\theta_s - \delta_i + \tau_k)]}$$

.

And the PCM can be expressed as:

$$P(X_{si} = x) = \frac{exp[\sum_{k=0}^{x}(\theta_s - \delta_i + \tau_{ki})]}{\sum_{h=0}^{m_i} exp[\sum_{k=0}^{h}(\theta_s - \delta_i + \tau_{ki})]}$$

Note the small but major difference - there is a unique rating scale structure for each item in the PCM while there is the same rating scale structure across items in the RSM.

These methods only apply when you have the same number of categories per items. if you have a different numbers of categories per item, to express the DIF models/work with formulas, one needs to work with the design matrices (see the polytomous DIF example in the `TAM` manual)

1.1 Using tam.mml.mfr to express the different models

To get a handle on using `tam.mml.mfr` to express different DIF models, let's first note how the different models (without DIF) can be expressed with formulas. The `TAM` manual notes that, when using `tam.mml`, setting `irtmodel = "RSM"` is equivalent to `item + step` and the partial credit model in `Conquest Parameterization` is equivalent to `item + item*step` `irtmodel = "PCM2"`. We can use formulas to express these two different models most easily.

For the default PCM model in TAM, `irtmodel = "PCM"`, the formula doesn't carry over quite as nicely to DIF models (at least transitioning between RSM and PCM), but we'll run it once and you can decide if you'd like to use it or not.

Let's test the equivalence of the formulas in `tam.mml.mfr` and specifying the IRT model in `tam.mml`.

1.1.1 Reading in the polytomous data and cleaning

```
#From lab 6
hls_poly <- read_csv("data/hls_poly.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double()
## )
```

```
## See spec(...) for full column specifications.
```

```
head(hls_poly)

# get the items
hls <- hls_poly %>%
  select(Hls1:Hls16)

# change everything to start with category 0
hls <- hls-1
```

```
# check out frequencies
apply(hls, 2, table)
```

```
##   Hls1 Hls2 Hls3 Hls4 Hls5 Hls6 Hls7 Hls8 Hls9 Hls10 Hls11 Hls12 Hls13 Hls14
## 0   63   76  129  160   65   90  140   50  146   139    47    44    52    64
## 1  204  207  176  148  166  196  162  154  144   163   116   138   176   181
## 2   44   32   11    7   78   26   12  101   21    13   130   119    83    60
## 3    6    2    1    2    8    5    3   12    6     2    24    16     6    12
##   Hls15 Hls16
## 0    52    69
## 1   148   185
## 2   105    57
## 3    12     6
```

```
#get facet
covs <- hls_poly %>%
  select(Gender)
```

1.1.2 Testing out the formulas

We'll express two formulas.

1. The RSM formula
2. The PCM2 formula
3. The default PCM formula

We'll run these via `tam.mml.mfr` and then run the equivalent, without the formula, in `tam.mml`.

1.2 Running RSM

Note, that `xsi.facets` from our model using the formula has the same rating scale structure and estimates (below; at the bottom of the table) as `item_irt` in `delta-tau` parameterization. There will be slight differences each time you estimate the model.

```
# Rating Scale Model
form_RSM <- ~item + step




set.seed(123)
facet_RSM <- tam.mml.mfr(hls, formulaA = form_RSM)

set.seed(123)
base_RSM <- tam.mml(hls, irtmodel = "RSM")
```

```
### note that these are the same with slight differences due to estimation only. ###

knitr::kable(facet_RSM$xsi.facets, caption = "RSM with facet formula")
```

Table 1: RSM with facet formula

parameter	facet	xsi	se.xsi
Hls1	item	1.5376772	0.1061042
Hls2	item	1.9185190	0.1087129
Hls3	item	2.8627063	0.1141930
Hls4	item	3.3041069	0.1173037
Hls5	item	1.1435353	0.1031617
Hls6	item	2.0855800	0.1097538
Hls7	item	2.9412779	0.1146789
Hls8	item	0.6707184	0.0996839
Hls9	item	2.8236662	0.1139588
Hls10	item	2.9412779	0.1146789
Hls11	item	0.1334125	0.0964204
Hls12	item	0.3593377	0.0976646

parameter	facet	xsi	se.xsi
Hls13	item	0.9961499	0.1020492
Hls14	item	1.2399960	0.1038921
Hls15	item	0.6508718	0.0995466
Hls16	item	1.4592933	0.1055320
step1	step	-3.0431164	0.0383237
step2	step	0.4158439	0.0417843
step3	step	2.6272725	0.0566977

```
knitr::kable(base_RSM$item_irt, caption = "RSM using base tam.mml")
```

Table 2: RSM using base tam.mml

item	alpha	beta	tau.Cat1	tau.Cat2	tau.Cat3
Hls1	1	1.5376300	-3.043111	0.4158464	2.627264
Hls2	1	1.9184719	-3.043111	0.4158464	2.627264
Hls3	1	2.8626591	-3.043111	0.4158464	2.627264
Hls4	1	3.3040595	-3.043111	0.4158464	2.627264
Hls5	1	1.1434880	-3.043111	0.4158464	2.627264
Hls6	1	2.0855329	-3.043111	0.4158464	2.627264
Hls7	1	2.9412307	-3.043111	0.4158464	2.627264
Hls8	1	0.6706709	-3.043111	0.4158464	2.627264
Hls9	1	2.8236191	-3.043111	0.4158464	2.627264
Hls10	1	2.9412307	-3.043111	0.4158464	2.627264
Hls11	1	0.1333648	-3.043111	0.4158464	2.627264
Hls12	1	0.3592901	-3.043111	0.4158464	2.627264
Hls13	1	0.9961025	-3.043111	0.4158464	2.627264
Hls14	1	1.2399487	-3.043111	0.4158464	2.627264
Hls15	1	0.6508243	-3.043111	0.4158464	2.627264
Hls16	1	1.4592460	-3.043111	0.4158464	2.627264

1.3 Running PCM2

Note that the interaction terms in (for instance, `Hls1:step1`) `facet_PCM2$xsi.facets` in the `facets`/formula version of the model is the same as the `tau` parameters (with slight differences due to estimation runs) in `base_PCM2$item_irt` which does not take the formula.

```
#Partial Credit Model 2- in this version, you'll also get a step main effect. You can sum the unique st
form_PCM2_main <- ~ item + item*step



#equivalent
```

```
form_PCM22_main <- ~item*step

#  without the step main effects - they're "absorbed" in the item-step interactions
form_PCM22 <- ~item + item:step


#### Run the Partial Credit Model ######
facet_PCM2 <- tam.mml.mfr(hls, formulaA = form_PCM2_main)

facet_PCM22 <- tam.mml.mfr(hls, formulaA = form_PCM22)

base_PCM2 <- tam.mml(hls, irtmodel = "PCM2")
```

Note that these are all the same, or when summed across items and categories, add up to the same value. That is, if you sum the facet formula with step main effects with the step values in table 3, you'll get something something similar to table 5 and 6 (slight estimation differences.)

```
knitr::kable(facet_PCM2$xsi.facets, caption = "facet: item+step+item*step")
```

Table 3: facet: item+step+item*step

parameter	facet	xsi	se.xsi
Hls1	item	1.4284666	0.1120166
Hls2	item	2.0743975	0.1198212
Hls3	item	2.9006026	0.1219610
Hls4	item	2.8094008	0.1195339
Hls5	item	1.1993422	0.1003686
Hls6	item	1.8194031	0.1132832
Hls7	item	2.4554444	0.1152087
Hls8	item	0.7815011	0.0989171
Hls9	item	2.0993126	0.1037181
Hls10	item	2.6305562	0.1165132
Hls11	item	0.3255012	0.0933676
Hls12	item	0.5134543	0.0975449
Hls13	item	1.1632081	0.1052322
Hls14	item	1.0705661	0.1003932
Hls15	item	0.7931028	0.0980850
Hls16	item	1.4350651	0.1057980
step1	step	-2.8971445	0.0380153
step2	step	0.6266464	0.0429058
step3	step	2.2704980	0.0573243
Hls1:step1	item:step	-0.3764517	0.1157783
Hls2:step1	item:step	-0.6780466	0.1139866
Hls3:step1	item:step	-0.3833432	0.1044942
Hls4:step1	item:step	0.2646442	0.1021601

parameter	facet	xsi	se.xsi
Hls5:step1	item:step	0.0145799	0.1147808
Hls6:step1	item:step	-0.0766898	0.1093388
Hls7:step1	item:step	0.2768040	0.1030261
Hls8:step1	item:step	-0.0194009	0.1172894
Hls9:step1	item:step	0.7907489	0.1022132
Hls10:step1	item:step	0.0842107	0.1033762
Hls11:step1	item:step	0.4664813	0.1128975
Hls12:step1	item:step	0.0667333	0.1171992
Hls13:step1	item:step	-0.3924903	0.1197987
Hls14:step1	item:step	0.0660206	0.1128452
Hls15:step1	item:step	0.0624761	0.1167942
Hls16:step1	item:step	-0.1662766	0.4308090
Hls1:step2	item:step	0.2441786	0.1179485
Hls2:step2	item:step	0.0645027	0.1249966
Hls3:step2	item:step	0.4367203	0.1432558
Hls4:step2	item:step	0.8900792	0.1481865
Hls5:step2	item:step	-0.4430975	0.1073200
Hls6:step2	item:step	0.4942605	0.1285544
Hls7:step2	item:step	0.6722268	0.1417289
Hls8:step2	item:step	-0.5105534	0.1030160
Hls9:step2	item:step	0.2648641	0.1327578
Hls10:step2	item:step	0.4387557	0.1407832
Hls11:step2	item:step	-0.7785723	0.0997947
Hls12:step2	item:step	-0.6258326	0.1009311
Hls13:step2	item:step	-0.4502271	0.1064097
Hls14:step2	item:step	0.0658499	0.1113514
Hls15:step2	item:step	-0.6083318	0.1026498
Hls16:step2	item:step	-0.1548230	0.4718161
Hls1:step3	item:step	0.1322731	0.1652770
Hls2:step3	item:step	0.6135439	0.1691659
Hls3:step3	item:step	-0.0533772	0.1773169
Hls4:step3	item:step	-1.1547234	0.1799886
Hls5:step3	item:step	0.4285176	0.1571376
Hls6:step3	item:step	-0.4175706	0.1687637
Hls7:step3	item:step	-0.9490308	0.1752183
Hls8:step3	item:step	0.5299543	0.1561061
Hls9:step3	item:step	-1.0556130	0.1675475

parameter	facet	xsi	se.xsi
Hls10:step3	item:step	-0.5229665	0.1746612
Hls11:step3	item:step	0.3120909	0.1506812
Hls12:step3	item:step	0.5590993	0.1546698
Hls13:step3	item:step	0.8427174	0.1602334
Hls14:step3	item:step	-0.1318705	0.1585345
Hls15:step3	item:step	0.5458557	0.1554923
Hls16:step3	item:step	0.3210996	0.6389106

```
knitr::kable(base_PCM2$item_irt, caption = "delta-tau")
```

Table 4: delta-tau

item	alpha	beta	tau.Cat1	tau.Cat2	tau.Cat3
Hls1	1	1.4284599	-3.273615	0.8708183	2.402797
Hls2	1	2.0745674	-3.575385	0.6909340	2.884451
Hls3	1	2.9029275	-3.282781	1.0602224	2.222558
Hls4	1	2.8101198	-2.633225	1.5153592	1.117866
Hls5	1	1.1993346	-2.882574	0.1835402	2.699034
Hls6	1	1.8193879	-2.973829	1.1209025	1.852926
Hls7	1	2.4554805	-2.620377	1.2987924	1.321585
Hls8	1	0.7815041	-2.916568	0.1160786	2.800490
Hls9	1	2.0992913	-2.106383	0.8915122	1.214871
Hls10	1	2.6310538	-2.813424	1.0646304	1.748793
Hls11	1	0.3255027	-2.430694	-0.1519153	2.582609
Hls12	1	0.5134593	-2.830463	0.0008037	2.829659
Hls13	1	1.1632000	-3.289659	0.1764170	3.113242
Hls14	1	1.0705679	-2.831142	0.6924792	2.138663
Hls15	1	0.7931058	-2.834692	0.0182993	2.816393
Hls16	1	1.4351682	-3.062963	0.4725020	2.590461

```
knitr::kable(base_PCM2$xsi, caption = "Base PCM2 XSI")
```

Table 5: Base PCM2 XSI

	xsi	se.xsi
Hls1	1.4284704	0.1120169
Hls2	2.0745816	0.1198221
Hls3	2.9029379	0.1219641

	xsi	se.xsi
Hls4	2.8101487	0.1195358
Hls5	1.1993449	0.1003690
Hls6	1.8193989	0.1132830
Hls7	2.4554921	0.1152086
Hls8	0.7815144	0.0989180
Hls9	2.0993027	0.1037174
Hls10	2.6310658	0.1165144
Hls11	0.3255122	0.0933689
Hls12	0.5134691	0.0975464
Hls13	1.1632103	0.1052327
Hls14	1.0705787	0.1003937
Hls15	0.7931162	0.0980860
Hls16	1.4351788	0.1057965
Hls1_step1	-3.2736157	0.1658666
Hls1_step2	0.8708197	0.1747539
Hls2_step1	-3.5753835	0.1607242
Hls2_step2	0.6909332	0.2005306
Hls3_step1	-3.2828749	0.1369375
Hls3_step2	1.0601233	0.3228042
Hls4_step1	-2.6332201	0.1318043
Hls4_step2	1.5153696	0.3951076
Hls5_step1	-2.8825746	0.1629729
Hls5_step2	0.1835419	0.1448096
Hls6_step1	-2.9738298	0.1484221
Hls6_step2	1.1209042	0.2163015
Hls7_step1	-2.6203802	0.1336788
Hls7_step2	1.2987925	0.3064704
Hls8_step1	-2.9165669	0.1704043
Hls8_step2	0.1160809	0.1347255
Hls9_step1	-2.1063836	0.1319171
Hls9_step2	0.8915146	0.2383484
Hls10_step1	-2.8134294	0.1344476
Hls10_step2	1.0646277	0.2972173
Hls11_step1	-2.4306948	0.1577130
Hls11_step2	-0.1519135	0.1277500
Hls12_step1	-2.8304629	0.1701291
Hls12_step2	0.0008058	0.1301600

	xsi	se.xsi
Hls13_step1	-3.2896597	0.1783867
Hls13_step2	0.1764182	0.1425965
Hls14_step1	-2.8311397	0.1575701
Hls14_step2	0.6924818	0.1552025
Hls15_step1	-2.8346900	0.1688962
Hls15_step2	0.0183017	0.1339095
Hls16_step1	-3.0629642	0.1616413
Hls16_step2	0.4725033	0.1599085

```r
knitr::kable(facet_PCM22$xsi.facets, caption = "No step main effects")
```

Table 6: No step main effects

parameter	facet	xsi	se.xsi
Hls1	item	1.4285020	0.1120172
Hls2	item	2.0746253	0.1198224
Hls3	item	2.9029964	0.1219645
Hls4	item	2.8102032	0.1195364
Hls5	item	1.1993743	0.1003692
Hls6	item	1.8194332	0.1132833
Hls7	item	2.4555376	0.1152090
Hls8	item	0.7815411	0.0989181
Hls9	item	2.0993373	0.1037177
Hls10	item	2.6311027	0.1165147
Hls11	item	0.3255375	0.0933689
Hls12	item	0.5134951	0.0975465
Hls13	item	1.1632411	0.1052329
Hls14	item	1.0706059	0.1003939
Hls15	item	0.7931429	0.0980861
Hls16	item	1.4352105	0.1057967
Hls1:step1	item:step	-3.2736261	0.1658666
Hls2:step1	item:step	-3.5754066	0.1607240
Hls3:step1	item:step	-3.2829114	0.1369388
Hls4:step1	item:step	-2.6332599	0.1318043
Hls5:step1	item:step	-2.8825840	0.1629729
Hls6:step1	item:step	-2.9738424	0.1484221
Hls7:step1	item:step	-2.6204026	0.1336788
Hls8:step1	item:step	-2.9165737	0.1704044

parameter	facet	xsi	se.xsi
Hls9:step1	item:step	-2.1063957	0.1319172
Hls10:step1	item:step	-2.8134395	0.1344476
Hls11:step1	item:step	-2.4306992	0.1577132
Hls12:step1	item:step	-2.8304682	0.1701292
Hls13:step1	item:step	-3.2896696	0.1783868
Hls14:step1	item:step	-2.8311472	0.1575701
Hls15:step1	item:step	-2.8346968	0.1688963
Hls16:step1	item:step	-3.0629745	0.1616414
Hls1:step2	item:step	0.8708130	0.1747545
Hls2:step2	item:step	0.6909138	0.2005315
Hls3:step2	item:step	1.0601083	0.3227909
Hls4:step2	item:step	1.5153369	0.3951142
Hls5:step2	item:step	0.1835358	0.1448099
Hls6:step2	item:step	1.1208958	0.2163024
Hls7:step2	item:step	1.2987759	0.3064723
Hls8:step2	item:step	0.1160763	0.1347258
Hls9:step2	item:step	0.8915060	0.2383495
Hls10:step2	item:step	1.0646240	0.2972199
Hls11:step2	item:step	-0.1519164	0.1277502
Hls12:step2	item:step	0.0008021	0.1301602
Hls13:step2	item:step	0.1764112	0.1425968
Hls14:step2	item:step	0.6924771	0.1552029
Hls15:step2	item:step	0.0182969	0.1339097
Hls16:step2	item:step	0.4724961	0.1599090
Hls1:step3	item:step	2.4028131	0.2409375
Hls2:step3	item:step	2.8844928	0.2569924
Hls3:step3	item:step	2.2228032	0.3506369
Hls4:step3	item:step	1.1179230	0.4165185
Hls5:step3	item:step	2.6990482	0.2180139
Hls6:step3	item:step	1.8529466	0.2623277
Hls7:step3	item:step	1.3216266	0.3343580
Hls8:step3	item:step	2.8004974	0.2172296
Hls9:step3	item:step	1.2148897	0.2724200
Hls10:step3	item:step	1.7488155	0.3262144
Hls11:step3	item:step	2.5826155	0.2029619
Hls12:step3	item:step	2.8296662	0.2142093
Hls13:step3	item:step	3.1132585	0.2283762

parameter	facet	xsi	se.xsi
Hls14:step3	item:step	2.1386702	0.2211703
Hls15:step3	item:step	2.8163999	0.2155406
Hls16:step3	item:step	2.5904785	0.2273737

1.4 Running PCM

Note that the parameterization below is actually the default parameterization in `tam` if you don't specify a model type and have polytomous data. However, I like using the parameterizations above because I think it's a bit easier to see what's going on when you're going between the models with DIF. Saying that, making use of it in the DIF models below is clarifying.

```
base_PCM <- tam.mml(hls, irtmodel = "PCM")
```

2 Polytomous DIF models

Now that we can create these different models using formulas in the `mfr` call, we can generate many different DIF models.

Among the types of models me way be interested in, we can check differential item functioning (checking the `delta` parameters across groups, but not the `taus`), checking the rating scale structures across groups, and checking both.

To see how we might run a few of these different models, let's go through a few options.

2.1 PCM - Basic DIF

2.1.1 Checking out delta parameters (item main effects, if you will)

Here, we'll just basicaly check if the deltas are the same. The nice part is that TAM gives us some extra output all the same.

This is something like a partial credit model with deltas interacted with covariates using default `TAM` PCM parameterization. You can run tests on the interactiosn between deltas and Gender if you like.

Note, if you run something like, `Gender*item + item*step`, you get a slightly different model, where you have step main effects as well.

```
form_delt_DIF <- ~ Gender*item + item:step

delt_dif <- tam.mml.mfr(hls, formulaA = form_delt_DIF, facets = covs)

form_delt_DIF1 <- ~ Gender*item + item*step

delt_dif1 <- tam.mml.mfr(hls, formulaA = form_delt_DIF1, facets = covs)
```

2.1.2 Differential category/rating scale functioning?

Above, we didn't check to see if there were any interactions between the categories and the items. So, I'll call this something like a "full-DIF" model. Note, you could instead use:

`form_PCM_full <- ~ Gender*item*step` and you will get the main effects of `step` parameters. This is probably something between a PCM and an RSM I'd imagine.

```
form_PCM_full <- ~ Gender*item+ item:step

delt_dif1 <- tam.mml.mfr(hls, formulaA = form_PCM_full, facets = covs)
```

2.2 Rating Scale DIF

```
form_rsm_dif <- ~ item*Gender + step


formRSMdelta <- tam.mml.mfr(hls, formulaA = form_rsm_dif, facets = covs)
```

Table 7: RSM with delta/item DIF

parameter	facet	xsi	se.xsi
Hls1	item	1.6091435	0.1062590
Hls2	item	1.9479596	0.1089233
Hls3	item	2.8764939	0.1144021
Hls4	item	3.3144011	0.1174913
Hls5	item	1.1577917	0.1033983
Hls6	item	2.1432270	0.1098879
Hls7	item	2.9287614	0.1149351
Hls8	item	0.7166940	0.0999522
Hls9	item	2.8377245	0.1141694
Hls10	item	2.8876132	0.1150682
Hls11	item	0.1556935	0.0966506
Hls12	item	0.3483272	0.0978608
Hls13	item	0.9587372	0.1022577
Hls14	item	1.2618452	0.1041288
Hls15	item	0.6056950	0.0997419
Hls16	item	1.4084369	0.1057625
step1	step	-3.0571966	0.0383736
step2	step	0.4167212	0.0418321
step3	step	2.6404754	0.0567667
Gender0	Gender	0.0108814	0.0265809
Gender1	Gender	-0.0108814	0.0265809
Hls1:Gender0	item:Gender	-0.2078940	0.0749603
Hls2:Gender0	item:Gender	-0.0752872	0.0758782
Hls3:Gender0	item:Gender	-0.0130143	0.0776603
Hls4:Gender0	item:Gender	0.0037223	0.0786059
Hls5:Gender0	item:Gender	-0.0381147	0.0739353
Hls6:Gender0	item:Gender	-0.1617397	0.0762021

parameter	facet	xsi	se.xsi
Hls7:Gender0	item:Gender	0.0766541	0.0778264
Hls8:Gender0	item:Gender	-0.1432029	0.0726441
Hls9:Gender0	item:Gender	-0.0143870	0.0775874
Hls10:Gender0	item:Gender	0.2206578	0.0778677
Hls11:Gender0	item:Gender	-0.0782442	0.0713466
Hls12:Gender0	item:Gender	0.0321171	0.0718292
Hls13:Gender0	item:Gender	0.1285567	0.0735149
Hls14:Gender0	item:Gender	-0.0605379	0.0742011
Hls15:Gender0	item:Gender	0.1503968	0.0725633
Hls16:Gender0	item:Gender	0.1803171	0.2910369
Hls1:Gender1	item:Gender	0.2078940	0.0749603
Hls2:Gender1	item:Gender	0.0752872	0.0758782
Hls3:Gender1	item:Gender	0.0130143	0.0776603
Hls4:Gender1	item:Gender	-0.0037223	0.0786059
Hls5:Gender1	item:Gender	0.0381147	0.0739353
Hls6:Gender1	item:Gender	0.1617397	0.0762021
Hls7:Gender1	item:Gender	-0.0766541	0.0778264
Hls8:Gender1	item:Gender	0.1432029	0.0726441
Hls9:Gender1	item:Gender	0.0143870	0.0775874
Hls10:Gender1	item:Gender	-0.2206578	0.0778677
Hls11:Gender1	item:Gender	0.0782442	0.0713466
Hls12:Gender1	item:Gender	-0.0321171	0.0718292
Hls13:Gender1	item:Gender	-0.1285567	0.0735149
Hls14:Gender1	item:Gender	0.0605379	0.0742011
Hls15:Gender1	item:Gender	-0.1503968	0.0725633
Hls16:Gender1	item:Gender	-0.1803171	0.2910369

2.2.1 Differential rating scale functioning?

What about checking if the rating scales are different? That is, what if we hold the item overall difficulties (or deltas) constant, but interact individual categories with with our facet.

```
form_rsm_rating <- ~ item + step*Gender


formRSMrating <- tam.mml.mfr(hls, formulaA = form_rsm_rating, facets = covs)
```

Table 8: RSM with rating scale DIF

parameter	facet	xsi	se.xsi
Hls1	item	1.5473879	0.1061564
Hls2	item	1.9285820	0.1087635
Hls3	item	2.8740717	0.1143085
Hls4	item	3.3165090	0.1174598
Hls5	item	1.1527969	0.1032308
Hls6	item	2.0958046	0.1098098
Hls7	item	2.9528069	0.1148018
Hls8	item	0.6792208	0.0997767
Hls9	item	2.8349539	0.1140707
Hls10	item	2.9528069	0.1148018
Hls11	item	0.1408193	0.0965233
Hls12	item	0.3672224	0.0977663
Hls13	item	1.0052021	0.1021262
Hls14	item	1.2493813	0.1039562
Hls15	item	0.6593371	0.0996402
Hls16	item	1.4689252	0.1055866
step1	step	-3.1055744	0.0383718
step2	step	0.4558108	0.0418231
step3	step	2.6497637	0.0567588
Gender0	Gender	-0.0225957	0.0265476
Gender1	Gender	0.0225957	0.0265476
step1:Gender0	step:Gender	0.1817711	0.0383718
step2:Gender0	step:Gender	-0.1270266	0.0418231
step3:Gender0	step:Gender	-0.0547445	0.0567588
step1:Gender1	step:Gender	-0.1817711	0.0383718
step2:Gender1	step:Gender	0.1270266	0.0418231
step3:Gender1	step:Gender	0.0547445	0.0567588

2.2.2 Rating Scale and Item DIF

If you are interested in interacting both the categories and the item overall difficulties, we can do this as well. Note, as the model gets more complicated this will take longer to run. It might be worth looking and plotting the deviances with each model run to make sure convergence looks ok.

```r
form_rsm_full <- ~ item + step*Gender



formRSMfull <- tam.mml.mfr(hls, formulaA = form_rsm_rating, facets = covs)
```

Table 9: RSM with rating_scale and item DIF

parameter	facet	xsi	se.xsi
Hls1	item	1.5473879	0.1061564
Hls2	item	1.9285820	0.1087635
Hls3	item	2.8740717	0.1143085
Hls4	item	3.3165090	0.1174598
Hls5	item	1.1527969	0.1032308
Hls6	item	2.0958046	0.1098098
Hls7	item	2.9528069	0.1148018
Hls8	item	0.6792208	0.0997767
Hls9	item	2.8349539	0.1140707
Hls10	item	2.9528069	0.1148018
Hls11	item	0.1408193	0.0965233
Hls12	item	0.3672224	0.0977663
Hls13	item	1.0052021	0.1021262
Hls14	item	1.2493813	0.1039562
Hls15	item	0.6593371	0.0996402
Hls16	item	1.4689252	0.1055866
step1	step	-3.1055744	0.0383718
step2	step	0.4558108	0.0418231
step3	step	2.6497637	0.0567588
Gender0	Gender	-0.0225957	0.0265476
Gender1	Gender	0.0225957	0.0265476
step1:Gender0	step:Gender	0.1817711	0.0383718
step2:Gender0	step:Gender	-0.1270266	0.0418231
step3:Gender0	step:Gender	-0.0547445	0.0567588
step1:Gender1	step:Gender	-0.1817711	0.0383718
step2:Gender1	step:Gender	0.1270266	0.0418231
step3:Gender1	step:Gender	0.0547445	0.0567588

3 PCM DIF in TAM when the item categories are not the same

What if we have some items with different response options. For instance, what if some items are dichotmous, some items have three categories and some items have four categories (or something). Unfortunately, we have to work with design matrices in TAM.

The A matrix basically specifies the category of a response - that is, it gives the response categories their, well, categories. In the model, the item difficulty parameters are multiplied by A matrix. The B matrix is known as the scoring matrix and describes how each category is to be scored and to which dimension. The B matrix multiplies the theta ability estimate. You can extract these objects from TAM output.

3.1 Exploring Design Matrices (these ar arrays)

```
# for item1 1
base_RSM$A[1,,]
```

```
##      Hls1 Hls2 Hls3 Hls4 Hls5 Hls6 Hls7 Hls8 Hls9 Hls10 Hls11 Hls12 Hls13 Hls14
## [1,]    0    0    0    0    0    0    0    0    0     0     0     0     0     0
## [2,]   -1    0    0    0    0    0    0    0    0     0     0     0     0     0
## [3,]   -2    0    0    0    0    0    0    0    0     0     0     0     0     0
## [4,]   -3    0    0    0    0    0    0    0    0     0     0     0     0     0
##      Hls15 Hls16 Cat1 Cat2
## [1,]     0     0    0    0
## [2,]     0     0   -1    0
## [3,]     0     0   -1   -1
## [4,]     0     0    0    0
```

```
# For item 2
base_PCM2$A[2,,]
```

```
##      Hls1 Hls2 Hls3 Hls4 Hls5 Hls6 Hls7 Hls8 Hls9 Hls10 Hls11 Hls12 Hls13 Hls14
## [1,]    0    0    0    0    0    0    0    0    0     0     0     0     0     0
## [2,]    0   -1    0    0    0    0    0    0    0     0     0     0     0     0
## [3,]    0   -2    0    0    0    0    0    0    0     0     0     0     0     0
## [4,]    0   -3    0    0    0    0    0    0    0     0     0     0     0     0
##      Hls15 Hls16 Hls1_step1 Hls1_step2 Hls2_step1 Hls2_step2 Hls3_step1
## [1,]     0     0          0          0          0          0          0
## [2,]     0     0          0          0         -1          0          0
## [3,]     0     0          0          0         -1         -1          0
## [4,]     0     0          0          0          0          0          0
##      Hls3_step2 Hls4_step1 Hls4_step2 Hls5_step1 Hls5_step2 Hls6_step1
## [1,]          0          0          0          0          0          0
## [2,]          0          0          0          0          0          0
## [3,]          0          0          0          0          0          0
## [4,]          0          0          0          0          0          0
##      Hls6_step2 Hls7_step1 Hls7_step2 Hls8_step1 Hls8_step2 Hls9_step1
## [1,]          0          0          0          0          0          0
## [2,]          0          0          0          0          0          0
## [3,]          0          0          0          0          0          0
## [4,]          0          0          0          0          0          0
##      Hls9_step2 Hls10_step1 Hls10_step2 Hls11_step1 Hls11_step2 Hls12_step1
## [1,]          0           0           0           0           0           0
## [2,]          0           0           0           0           0           0
## [3,]          0           0           0           0           0           0
## [4,]          0           0           0           0           0           0
##      Hls12_step2 Hls13_step1 Hls13_step2 Hls14_step1 Hls14_step2 Hls15_step1
## [1,]           0           0           0           0           0           0
## [2,]           0           0           0           0           0           0
## [3,]           0           0           0           0           0           0
## [4,]           0           0           0           0           0           0
##      Hls15_step2 Hls16_step1 Hls16_step2
## [1,]           0           0           0
## [2,]           0           0           0
## [3,]           0           0           0
## [4,]           0           0           0
```

```
# For item 1
formRSMrating$A[1, , ]
```

```
##        Hls1 Hls2 Hls3 Hls4 Hls5 Hls6 Hls7 Hls8 Hls9 Hls10 Hls11 Hls12 Hls13
## _step0    0    0    0    0    0    0    0    0    0     0     0     0     0
## _step1   -1    0    0    0    0    0    0    0    0     0     0     0     0
## _step2   -2    0    0    0    0    0    0    0    0     0     0     0     0
## _step3   -3    0    0    0    0    0    0    0    0     0     0     0     0
##        Hls14 Hls15 Hls16 step1 step2 Gender0 step1:Gender0 step2:Gender0
## _step0     0     0     0     0     0       0             0             0
## _step1     0     0     0    -1     0      -1            -1             0
## _step2     0     0     0    -1    -1      -2            -1            -1
## _step3     0     0     0     0     0      -3             0             0
```

We have to alter these if we want to get coherent output for a model to test for DIF in polytomous data with different number of resposne options across items.

This is just using the TAM example case. What we'll do here is (after the usual data cleaining) 1. use `designMatrices.mfr2` from `TAM` which will generate the design matrices and other information needed to run the model properly. It'll be stored in an object called `des2`.
2. Extract an altered data set that creates "pseudo items" so that each item/step-country pair is a unique item. So, for step 0, item 1 - there are two items. One for if you're in country 1 and one for country 2. If you're in country 2, the item for country 1 is listed as missing. And vice versa.
3. TAM uses sum constraints - so we need to eliminate one item for identification basically - they're redundant.
4. Create the A and B matrices appropriately.

The example below uses `item*step + item*cnt`. You may wish to alter this DIF formula to your interests.

```
# get data from TAM
data(data.timssAusTwn.scored)

dat <- data.timssAusTwn.scored

# extract item response data
resp <- dat[, sort(grep("M", colnames(data.timssAusTwn.scored), value=TRUE)) ]


# some descriptives
psych::describe(resp)
```

```
##          vars    n mean   sd median trimmed mad min max range  skew kurtosis
## M032166     1 1769 0.70 0.46      1    0.74   0   0   1     1 -0.85    -1.28
## M032595     2 1769 0.68 0.47      1    0.73   0   0   1     1 -0.78    -1.40
## M032626     3 1769 0.61 0.49      1    0.63   0   0   1     1 -0.44    -1.81
## M032673     4 1769 0.68 0.47      1    0.72   0   0   1     1 -0.75    -1.44
## M032692     5 1769 0.75 0.95      0    0.69   0   0   2     2  0.52    -1.69
## M032721     6 1769 0.51 0.50      1    0.51   0   0   1     1 -0.04    -2.00
## M032757     7 1769 1.41 0.89      2    1.52   0   0   2     2 -0.91    -1.11
## M032760A    8 1769 1.04 0.98      2    1.05   0   0   2     2 -0.07    -1.96
## M032760B    9 1769 0.38 0.49      0    0.36   0   0   1     1  0.47    -1.78
## M032760C   10 1769 0.32 0.47      0    0.28   0   0   1     1  0.76    -1.43
## M032761    11 1769 0.72 0.88      0    0.65   0   0   2     2  0.58    -1.45
##            se
## M032166  0.01
## M032595  0.01
## M032626  0.01
## M032673  0.01
## M032692  0.02
## M032721  0.01
## M032757  0.02
## M032760A 0.02
## M032760B 0.01
## M032760C 0.01
## M032761  0.02
```

```
# define facets: 'cnt' is group identifier
facets <- data.frame( "cnt"=dat$IDCNTRY)


# create design matrices - this is a key step.
des2 <- designMatrices.mfr2( resp=resp, facets=facets,
                   formulaA=~item*step + item*cnt)



# restructured data set: pseudoitem=item x country- this creates new items where each step and each ite
# missing if in cnt2. Second item is if you're in cnt2 and what you selected for original item 1)
resp2 <- des2$gresp$gresp.noStep


# A design matrix
A <- des2$A$A.3d

# redundant xsi parameters must be eliminated from design matrix - basically these are the first steps
# sum constraint to zero means you can eliminate one step from each item.
xsi.elim <- des2$xsi.elim

A <- A[,, - xsi.elim[,2] ]

# extract loading matrix B
B <- des2$B$B.3d
```

```
# estimate model
mod1 <- TAM::tam.mml( resp=resp2, A=A, B=B, control=list(maxiter=100) )
```

```
summary(mod1)
```

```
## ------------------------------------------------------------
## TAM 3.5-19 (2020-05-05 22:45:39)
## R version 3.6.0 (2019-04-26) x86_64, mingw32 | nodename=LAPTOP-K7402PLE | login=katzd
##
## Date of Analysis: 2021-03-07 09:18:45
## Time difference of 1.203952 secs
## Computation time: 1.203952
##
## Multidimensional Item Response Model in TAM
##
## IRT Model: 1PL
## Call:
## TAM::tam.mml(resp = resp2, A = A, B = B, control = list(maxiter = 100))
##
```

```
## ------------------------------------------------------------
## Number of iterations = 91
## Numeric integration with 21 integration points
##
## Deviance = 20473.62
## Log likelihood = -10236.81
## Number of persons = 1769
## Number of persons used = 1769
## Number of items = 22
## Number of estimated parameters = 27
##     Item threshold parameters = 26
##     Item slope parameters = 0
##     Regression parameters = 0
##     Variance/covariance parameters = 1
##
## AIC = 20528  | penalty=54    | AIC=-2*LL + 2*p
## AIC3 = 20555  | penalty=81    | AIC3=-2*LL + 3*p
## BIC = 20676  | penalty=201.91    | BIC=-2*LL + log(n)*p
## aBIC = 20590  | penalty=116.07    | aBIC=-2*LL + log((n-2)/24)*p  (adjusted BIC)
## CAIC = 20703  | penalty=228.91    | CAIC=-2*LL + [log(n)+1]*p  (consistent AIC)
## AICc = 20528  | penalty=54.87    | AICc=-2*LL + 2*p + 2*p*(p+1)/(n-p-1)  (bias corrected AIC)
## GHP = 0.52746     | GHP=( -LL + p ) / (#Persons * #Items)  (Gilula-Haberman log penalty)
##
## ------------------------------------------------------------
## EAP Reliability
## [1] 0.805
## ------------------------------------------------------------
## Covariances and Variances
##       [,1]
## [1,] 2.294
## ------------------------------------------------------------
## Correlations and Standard Deviations (in the diagonal)
##       [,1]
## [1,] 1.515
## ------------------------------------------------------------
## Regression Coefficients
##      [,1]
## [1,]    0
## ------------------------------------------------------------
## Item Parameters -A*Xsi
##                item    N     M xsi.item AXsi_.Cat1 AXsi_.Cat2 B.Cat1.Dim1
## 1   M032166-cnt_158  719 0.736   -1.473     -1.473         NA           1
## 2   M032166-cnt_36_ 1050 0.668   -0.991     -0.991         NA           1
## 3  M032760C-cnt_158  719 0.566   -0.372     -0.372         NA           1
## 4  M032760C-cnt_36_ 1050 0.156    2.296      2.296         NA           1
## 5   M032761-cnt_158  719 1.277   -0.714     -0.570     -1.427           1
## 6   M032761-cnt_36_ 1050 0.333    1.924      2.067      3.847           1
## 7   M032595-cnt_158  719 0.844   -2.394     -2.394         NA           1
## 8   M032595-cnt_36_ 1050 0.570   -0.422     -0.422         NA           1
## 9   M032626-cnt_158  719 0.883   -2.831     -2.831         NA           1
## 10  M032626-cnt_36_ 1050 0.418    0.430      0.430         NA           1
## 11  M032673-cnt_158  719 0.819   -2.152     -2.152         NA           1
## 12  M032673-cnt_36_ 1050 0.577   -0.465     -0.465         NA           1
## 13  M032692-cnt_158  719 1.433   -1.068      0.736     -2.135           1
## 14  M032692-cnt_36_ 1050 0.280    1.945      3.749      3.890            1
## 15  M032721-cnt_158  719 0.584   -0.481     -0.481         NA            1
## 16  M032721-cnt_36_ 1050 0.459    0.196      0.196         NA            1
## 17  M032757-cnt_158  719 1.723   -2.084     -0.359     -4.168            1
## 18  M032757-cnt_36_ 1050 1.202   -0.504      1.221     -1.008            1
## 19 M032760A-cnt_158  719 1.555   -1.438      0.577     -2.877            1
## 20 M032760A-cnt_36_ 1050 0.682    0.667      2.683      1.334            1
## 21 M032760B-cnt_158  719 0.609   -0.633     -0.633         NA            1
## 22 M032760B-cnt_36_ 1050 0.231    1.643      1.643         NA            1
##    B.Cat2.Dim1
## 1            0
## 2            0
## 3            0
## 4            0
## 5            2
## 6            2
## 7            0
## 8            0
## 9            0
## 10           0
## 11           0
## 12           0
## 13           2
## 14           2
## 15           0
## 16           0
## 17           2
## 18           2
## 19           2
## 20           2
## 21           0
## 22           0
##
## Item Parameters Xsi
##                     xsi se.xsi
## M032166          -2.654  0.061
## M032595          -2.830  0.062
## M032626          -2.623  0.065
## M032673          -2.731  0.062
## M032692           0.439  0.045
## M032721          -1.565  0.057
## M032757          -1.294  0.039
## M032760A         -0.386  0.040
## M032760B         -0.917  0.063
## M032760C         -0.460  0.067
## M032761           0.605  0.045
## step1             1.422  0.021
## cnt_158          -1.016  0.016
## M032692:step1     0.381  0.062
## M032757:step1     0.303  0.060
## M032760A:step1    0.594  0.062
## M032166:cnt_158   0.775  0.036
## M032595:cnt_158   0.031  0.036
## M032626:cnt_158  -0.614  0.037
## M032673:cnt_158   0.173  0.036
## M032692:cnt_158  -0.490  0.032
## M032721:cnt_158   0.678  0.035
## M032757:cnt_158   0.226  0.030
## M032760A:cnt_158 -0.036  0.030
## M032760B:cnt_158 -0.122  0.036
## M032760C:cnt_158 -0.318  0.037
##
## Item Parameters in IRT parameterization
##                   item alpha   beta tau.Cat1 tau.Cat2
## 1   M032166-step0-cnt1     1 -1.473       NA       NA
## 2   M032166-step0-cnt2     1 -0.991       NA       NA
## 3  M032760C-step0-cnt1     1 -0.372       NA       NA
## 4  M032760C-step0-cnt2     1  2.296       NA       NA
## 5   M032761-step0-cnt1     1 -0.714    0.144   -0.144
## 6   M032761-step0-cnt2     1  1.924    0.144   -0.144
## 7   M032595-step0-cnt1     1 -2.394       NA       NA
## 8   M032595-step0-cnt2     1 -0.422       NA       NA
## 9   M032626-step0-cnt1     1 -2.831       NA       NA
## 10  M032626-step0-cnt2     1  0.430       NA       NA
## 11  M032673-step0-cnt1     1 -2.152       NA       NA
## 12  M032673-step0-cnt2     1 -0.465       NA       NA
## 13  M032692-step0-cnt1     1 -1.068    1.803   -1.803
## 14  M032692-step0-cnt2     1  1.945    1.803   -1.803
## 15  M032721-step0-cnt1     1 -0.481       NA       NA
## 16  M032721-step0-cnt2     1  0.196       NA       NA
## 17  M032757-step0-cnt1     1 -2.084    1.726   -1.726
## 18  M032757-step0-cnt2     1 -0.504    1.726   -1.726
## 19 M032760A-step0-cnt1     1 -1.438    2.016   -2.016
## 20 M032760A-step0-cnt2     1  0.667    2.016   -2.016
## 21 M032760B-step0-cnt1     1 -0.633       NA       NA
## 22 M032760B-step0-cnt2     1  1.643       NA       NA
```

```
# The sum of all DIF parameters is set to zero. The DIF parameter for the last
# item is therefore obtained
xsi1 <- mod1$xsi
```